PLASMA-Z 2008 Team Description Paper

Aphiwat Kriengwattanakul[2], Sikornron Wattanasarn[2],
Sompol Suntharasantic[2] ,Sarut Chanpichaigosol[2],
Chanyuth Kokaewwichian[3], Theeraphol Wattanavekin[3],
Prinya Ovatlarnporn[3], Suphan Phayong[3], Nattapol Tongoa[3],
Teerapap Changwichukarn[3], Tanakorn Leesatapornwongsa[3]
Wittaya Wannasuphoprasit[1], Manop Wongsaisuwan[2]

Department of Mechanical Engineering[1]
Department of Electrical Engineering[2]
Department of Computer Engineering[3]
Chulalongkorn University
Phayathai Road, Bangkok, Thailand, 10330
http://eic.linux.eng.chula.ac.th
parnurzeal@gmail.com , prinny16@hotmail.com , tanakorn.lsv@gmail.com

Abstract. Robots that can play football efficiently as a team must consist of four main parts: mechanical part, electrical part, vision part, and AI part. Making entire system to work properly is a fundamental problem. Each part has to be well experimented and designed for making them work correctly both when isolated and when integrated with others.

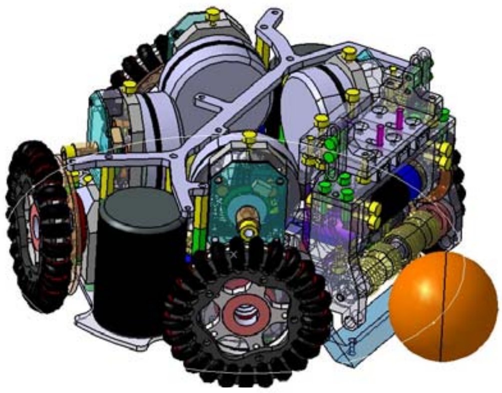
1 Introduction

Plasma-Z is a robot soccer team from Engineering Inventor Club, Faculty of Engineering, Chulalongkorn University, Thailand. Plasma-Z has joined Robocup Small Size League since 2003. So far, Plasma-Z has won Robocop Thailand Championship in 2003, 2004, 2005, 2006, 2007 and 2008. Especially, Plasma-Z won the second place in Small-Size Robot League at Atlanta, USA, 2007 and the third place in Small-Size Robot League and the first place in Small-Size Robot League Technical Challenge at Bremen, Germany, 2006. This year we are experimenting and developing many parts of our system for even better performance to join World Robocop 2008.

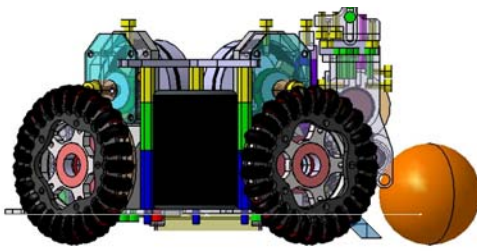
2 Mechanical system

Our robots are equipped with 4 omni-directional wheels. Each is driven by a redundant brushless Maxon EC-45 Flat 30-watt motor, with 7:1 external spur gear and 300-tick encoder. Theoretically, our robot can run with maximum velocity 2.49 m/s and increase speed with maximum acceleration 11.62 m/s^2. The velocity and acceleration profiles of our robot are shown in Figure 3 and Figure 4 respectively. To manipulate the ball, we use a horizontal dribbler system which is designed as two linear slide suspensions and two axles used for spinning the ball with one DOF in total. The types of material for spinning are rubber, foam and silicone.

Lastly, the third system is the shooting system. The conceptual design is to create two separate kicking mechanisms. One is for flat-shooting and the other is designed for chip kick. We designed chip kick mechanism like a swinging arm with one fixed axle and one ferrite core or steel rod moving to hit the swinging arm in order to shoot the ball. As a result, this chip kick mechanism can shoot ball more than 60 cm in height and 1 meter in length.



Robot height 145 mm
Robot diameter 178 mm
Percentage of ball coverage 20%



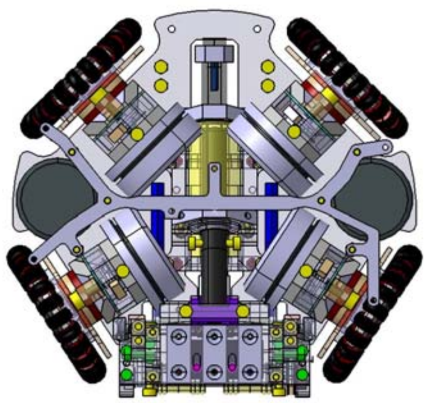


Fig. 2 Top view and side view to show the height of the robot, the maximum diameter of its projection to the ground, and the maximum percentage of ball coverage

V[m/s]

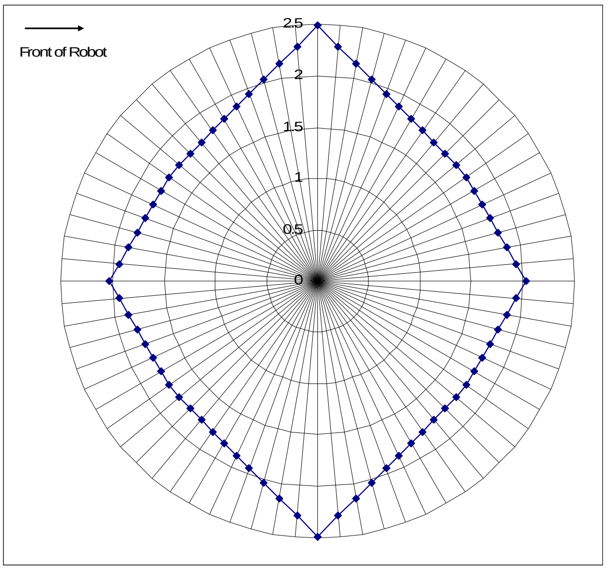


Fig. 3. Velocity profile of robot: maximum speed = 2.49 m/s, average speed = 1.91 m/s and minimum speed = 1.76 m/s.

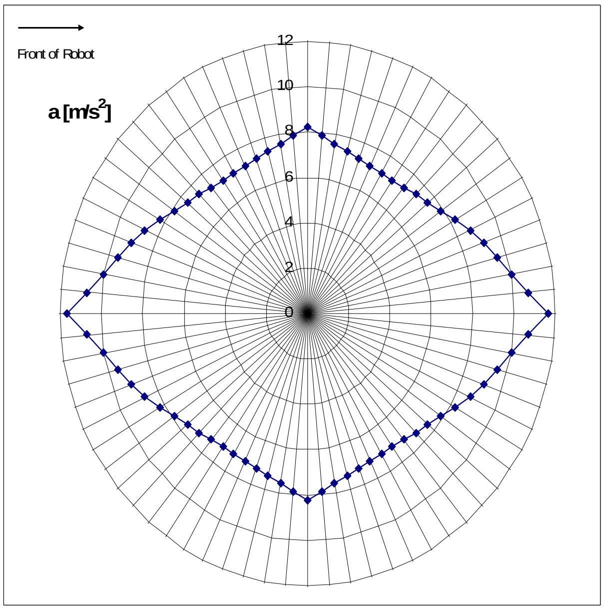


Fig. 4. Acceleration profile of robot: maximum acceleration = 11.62 m/s^2, average acceleration = 8.49 m/s^2 and minimum acceleration = 7.51 m/s^2.

3 Electrical System

Our main circuit boards use FPGA to implement an MCU and local feedback control loop. About motor driving, we use MOSFET circuit to drive brushless DC motor (Maxon). The motors speed is controlled by the MCU and logic circuit (feedback control loop) which is synthesized on FPGA. We are busy developing the gyro and electronic compass unit. Also, we use LINX TXM-900-HP3-PPS and RXM-900-HP3-PPS for wireless communication in frequencies around 900 MHz. Furthermore, we try to design a new circuit board that places the motor drive circuit, wireless communication part, and main control circuit in a single board. The separated shooting circuit board will connect to the main circuit as usual.

3.1 Central Control

We use an FPGA with embedded soft microcontroller as the main controller to generate all control signals for all parts. It has to cope with full duplex wireless communication via radio frequency to get the command packet from the AI computer at 60 times per second. Once the packets arrive, they have to be decoded and processed along with the data from four hall sensors, four encoders mounted with motors, one gyroscope, and one accelerometer to generate signals to be sent to motor drivers, shooting system, and others.

3.2 Motor Driver

This task of the circuit is to drive 4 brushless DC motor for wheels and 1 brush DC motor for dribbler. We have developed our driver circuit to drive brushless DC motor efficiently. We use 3 p-channel mosfet, 3 n-channel mosfet and 1 open-collector buffer to drive 1 motor according to the signal sent from central control circuit. Moreover, it has to get the hall sensor signal from motor and change it into 3.3 volt before send it back to main controller.

3.3 Shooting System

Most parts are essentially the same as in the previous year. There are two kickers: flat kick and chip kick. We develop the flat kick system to shoot at a velocity of approximately 10 m/s with two capacitors; 2200uF and 250 V. Method for calibrating the uncertainty of kicking distance is being invented. We are still finding some materials that have better permeability for a better energy conversion. Moreover, as for its conversion, we also search for a new type of curved solenoid which moves on the arc of a circle.

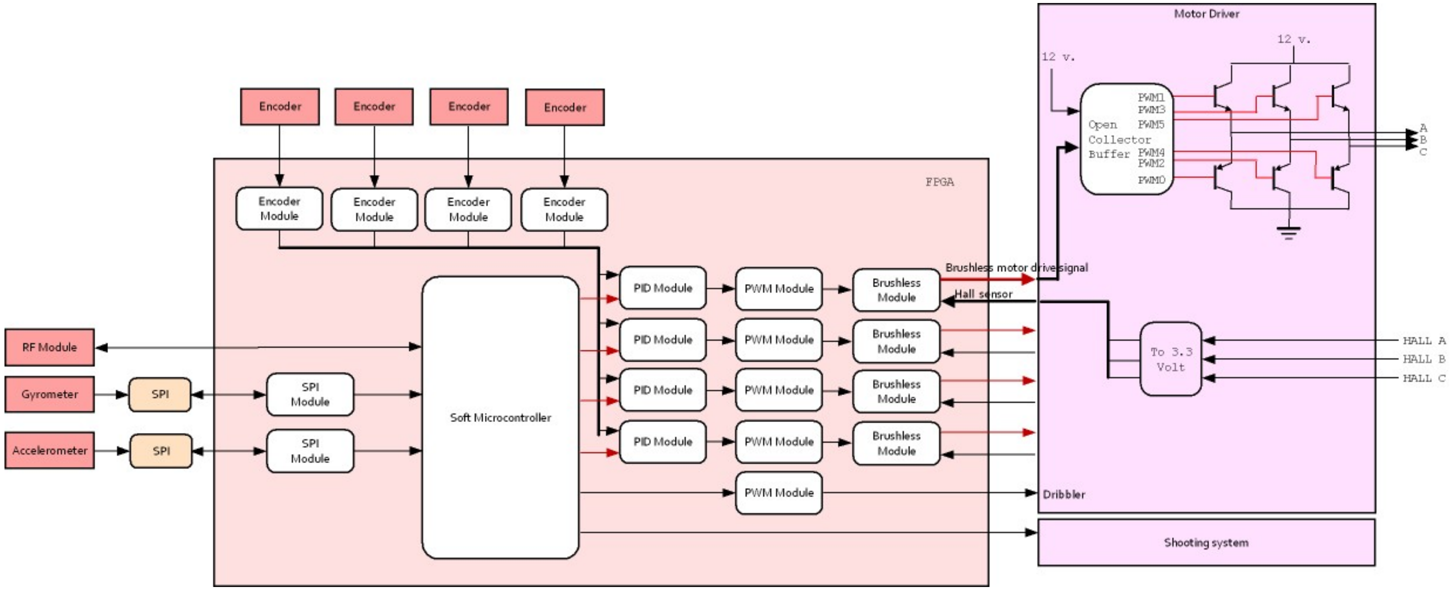


Fig. 4. Schematic diagram

4 Vision system

The vision system consists of a high speed IEEE1394 PCI Card running in a PC. The processing is all done via software. The vision system delivers the coordinates of all players in the field and the ball. It is capable of identifying individual player of the team by means by different pattern-coded figure of each player. It also distinguishes between our players and the opponent players. We achieve the frame rate at 60 frames/second.

4.1 Hardware

For each half of the field, there is one master CCD camera connected with IEEE 1394 PCI Card at the resolution of 640x480 pixels running at 400 M bits/second transfer rate. This PCI Card is hosted inside a PC. There is no special image processing hardware, all processing are done via software running in the PC.

4.2 Calibration

To separate the robotic players from the background, a threshold is established. The calibration procedure is necessary to establish this threshold. The whole viewing area is divided into tiles and in each tile the threshold will be set separately. This technique is used to battle the non-uniformity of the lighting in the playing field. There are other geometric distortions from the camera lens. The camera is mounted overhead and is assumed to face towards the field perpendicularly. The origin coordinate of the camera must be established in relation to the image from the real world. Both rotational and translational transformations are calibrated. The lens modeling is used to cancel lens distortion. The whole calibration procedure is time consuming hence it is done only once in the beginning of the game.

4.3 Object Localization

The low level image processing is done using both color and intensity. Objects are detected by color segmentation. The raw “blob” objects are then passed through filtering process. The filtering process rejects erroneous objects and unrecognizable objects. The remaining objects are tagged with coordinates and other attributes to facilitate further analysis. An individual player is recognized by detecting its marker location and the motion history. To detect a marker, the robot pattern skeleton is used. By decoding the pattern of marker, each robot is given its identification. If the object having that pattern is not identified as our player, it is assumed to be the opponent. The attribute of an object includes its location and orientation.

The ball is the most difficult object to be detected reliably especially when it is in motion. For early processing, the color segmentation produces candidates, which are then selected according to heuristic rules. Special rules are applied for specific situations such as when the ball is in contact with a robot. All object information is then sent to Intelligence module to plan and perform team action

5 AI System

AI, whose duty is to receive data from the vision session for processing before transferring to command all robots, consists of

5.1 Vision Filtering

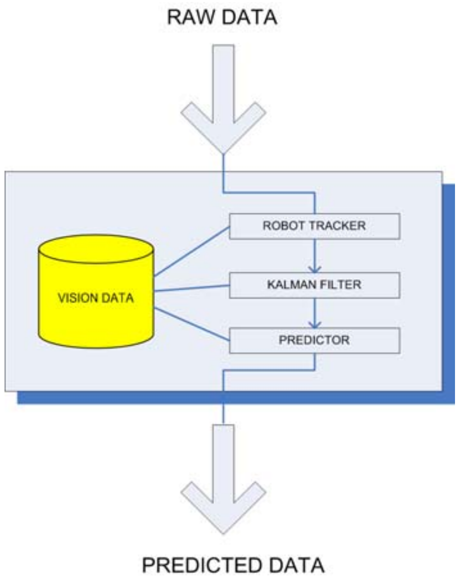


Fig. 5. Vision filtering block diagram

The main feature of Vision filtering is to receive data from the vision part. Because the raw data from two cameras consists of position and direction of our robots, position of opponent robots and position of the ball is delayed, the vision filtering session is designed to collect and analyze the data to determine the actual position. For this purpose, Kalman's predictor is used. Furthermore, the past data are used to determine other useful information such as the direction of robots and ball. Finally, the overall data is sent to the next part.

5.2 Strategy

Strategy is considered the most important part of AI. It is the brain for planning strategy and coordination among robots in both attack mode and defense mode. A layer architecture is used in this part.

Manager. Manager performs like a human football team manager. By receiving data like game score, ball possession, opponent strategy and referee box signal, manager applies the most suitable play for that moment.

Play. Play is a strategy of AI which consists of a specific game plan. Play will determine play form to all robots by assigning role to robot and select zone for robot. Play may issue some command to robot role such as ball possession, select role play, force role to passing ball to another robot or shooting ball. The first robot is ball possessor role and the other robots are the guardians. When the first robot cannot control ball anymore, he can pass a ball to left or right robots.

Group. Group is a group of robots collaborating in a mission such as attacking group of three robots or defensive group of two robots**.**

Role. Role is performed as robot behavior which is assigned by Play to control robot to performed a specific action such as manipulate ball to zone, run to zone, scramble ball from opponent, get ball. The Role mechanism first assigns a particular skill to the robot then generates the best point for robot action and set another parameter to Skill.

Skill. Skill is a set of basic knowledge for every robot, such as how to move to a point, how to get the ball and kick. Skill module generates path (a set of points), dribbling and kicking commands that will be proceeded by varying of trajectory module which selected by Skill module. Each of skills has different main idea of generating path for robot, "get the ball skill" is different from "move to point skill" in many ways. We can study and test each skill independently for the best performance for each of skills.

Trajectory. Trajectory is a set of method that generates velocities to control robot. For the need of skills, some require accurate position; some require fast motion, but fast motion has a trade off with slipping. So each of trajectories methods are created to serve varying need of skills.

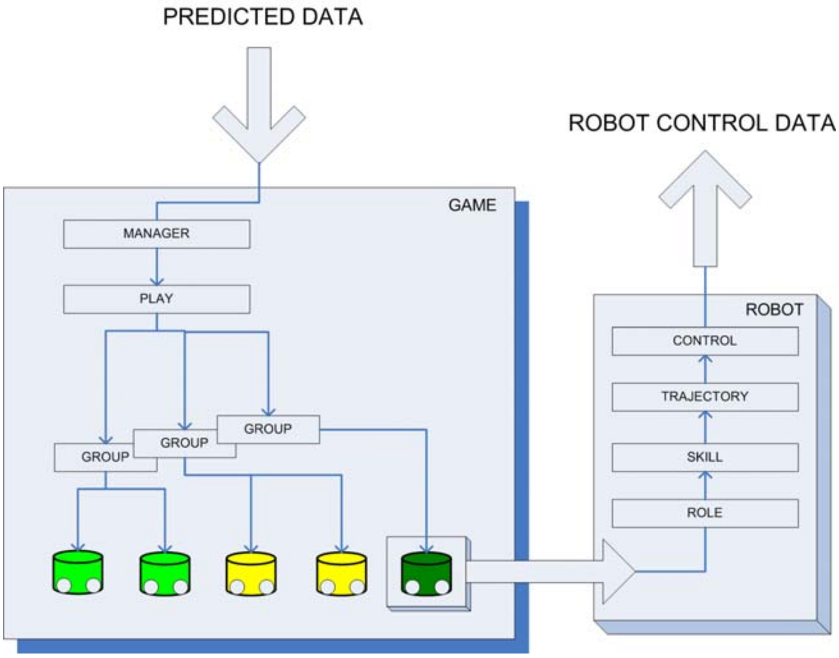


Fig. 6. Strategy layer architecture

Control. Control is the lowest layer of AI which controls robot movement. Control receives control data such as velocity, angular velocity from Trajectory. In the topic about vision data processing, we have already introduced Kalman's and Smith's predictors that make Intelligence system process more efficiently.

Learning System. AI System can learn trajectory and control by itself. Learning system applies Artificial Neural Network (ANN) for detect error which may cause from mechanics or others and corrects them. Robots can move and receive a ball more accuracy if AI system have already detected error in the past and put it in neural network. Moreover, AI System can learn in upper level such as "play" by ANN too. So the robots team may have different behavior in same situation if it has already learnt some events from the situation in the past.

5.3 Simulation System

We have developed a new robot simulator as another application that is completely independent from intelligence system. It is capable of simulating all the physical properties and delay latency of overall system. The physical properties of the field, robots and ball are done by physics engine library called Open Dynamics Engine (ODE) [1]. The new simulator provides connection socket for two intelligence systems which means the simulator is able to simulate a real playing situation of two teams and could provide a lot of benefits in evolving team strategy for real competition situation.

References

[1] Open Dynamics Engine, "An open source, high performance library for simulating rigid body dynamics", http://ode.org, August 2006